

**ΘΕΜΑ 1.**

A) Ποιος είναι ο βασικός λόγος για τον οποίο στοχεύουμε να έχουμε ισοζυγισμένα δένδρα αναζήτησης. B) Να εισάγετε τα ακόλουθα στοιχεία με τη σειρά που δίνονται σε ένα αρχικά άδειο δένδρο AVL και να εντοπίσετε τις περιστροφές (απλές και διπλές): 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12. Γ) Πόσες και ποιές διασχίσεις είναι απαραίτητες ώστε να μπορέσουμε να κατασκευάσουμε το αρχικό δυαδικό δένδρο; Να διατυπώσετε έναν αλγόριθμο σε ψευδοκώδικα που να παίρνει στην είσοδο τις διασχίσεις και θα δίνει στην έξοδο το δυαδικό δένδρο που αντιστοιχεί σε αυτές.

**ΘΕΜΑ 2.**

A) Να δώσετε μία έκφραση για τον μέσο αριθμό των ελέγχων (probes) που απαιτούνται για μη επιτυχή αναζήτηση στον κατακερματισμό με ανοικτή διεύθυνση και να δικαιολογήσετε την απάντησή σας. Σχολιάστε την επίδοση της μεθόδου σε σχέση με τον παράγοντα φόρτωσης (load factor) του πίνακα κατακερματισμού. B) Δίνονται  $n$  ακέραιοι αριθμοί. Χρησιμοποιώντας κατακερματισμό να περιγράψετε έναν αλγόριθμο απαλοιφής των διπλών εμφανίσεων των αριθμών. Αν δεν υπήρχε η δυνατότητα χρήσης κατακερματισμού πως θα λύνατε το πρόβλημα;

**ΘΕΜΑ 3.**

Δίνεται το μη κατευθυνόμενο γράφημα  $(1,2,10), (1,3,5), (2,4,1), (2,5,2), (1,5,20), (3,4,4), (4,5,2), (5,6,5), (\dots)$  όπου σε κάθε τριάδα οι δύο πρώτοι αριθμοί δηλώνουν τις κορυφές που συνδέονται και ο τρίτος αριθμός δηλώνει το βάρος της ακμής. A) Να δώσετε το δένδρο DFS για το γράφημα. B) Να εφαρμόσετε τον αλγόριθμο εύρεσης ελάχιστων μονοπατιών από την κορυφή 5 προς όλες τις άλλες και να δώσετε το κόστος της κάθε διαδρομής. Να διατυπώσετε τον αλγόριθμο σε ψευδοκώδικα.

**ΘΕΜΑ 4.**

A) Δίνεται σύνολο από  $n$  ακέραιους αριθμούς. Να προσδιορίσετε την πολυπλοκότητα κατασκευής ενός σωρού ελαχίστων για τους  $n$  αριθμούς και να δικαιολογήσετε την απάντησή σας. B) Να περιγράψετε τον αλγόριθμο ταξινόμησης σωρού (heapsort) και να δώσετε την πολυπλοκότητα χειρότερης περίπτωσης δικαιολογώντας την απάντησή σας. Ο αλγόριθμός είναι βέλτιστος ή όχι; Να δώσετε επίσης την πολυπλοκότητα χειρότερης περίπτωσης για όσους αλγορίθμους ταξινόμησης γνωρίζετε.

**ΘΕΜΑ 5.**

Έστω η αριθμητική έκφραση  $A * B + (C + D) / E - F$ . A) Να μετατρέψετε την έκφραση σε postfix αναπαράσταση χρησιμοποιώντας τη δομή δεδομένων της στοίβας και να διατυπώσετε τον αλγόριθμο μετατροπής σε ψευδοκώδικα. B) Πως θα λύνατε το πρόβλημα αν σας έδιναν ένα δένδρο έκφρασης για την πιο πάνω έκφραση;

Τα θέματα είναι ισοδύναμα.  
Διάρκεια εξέτασης 3 ώρες. Η σελίδα αυτή να παραδοθεί.  
Καλή επιτυχία!

**Λύσεις Θεμάτων Ιανουαρίου 2010**  
**ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ**

**Θέμα 1**

A) Ο βασικός λόγος που στοχεύουμε να έχουμε ισοζυγισμένα δέντρα αναζήτησης είναι για να πετυχαίνουμε λογαριθμικό χρόνο  $O(\log n)$  στην αναζήτηση ενός στοιχείου. Εάν το δέντρο δεν ήταν ισοζυγισμένο, η δομή του θα εξαρτιόταν από την σειρά εισαγωγής των στοιχείων και τα κλαδιά του θα μεγάλωναν πολύ, ώστε στην χειρότερη περίπτωση να είχαμε  $O(n)$  στην αναζήτηση.

B) όπως [θέμα 3, Ιανουαρίου 2011](#)

Γ) [παράδειγμα λύσης](#)

**Θέμα 2**

A) όπως [θέμα 1, Ιανουαρίου 2011](#)

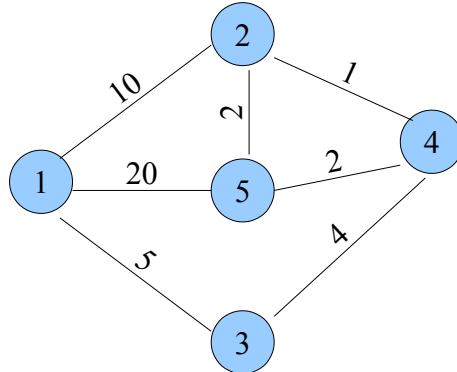
B) όπως [θέμα 1B, Ιανουαρίου 2011](#), περίπτωση της ένωσης.

Εάν δεν είχα κατακερματισμό, θα μπορούσα να φτιάξω ένα AVL και όταν πέφτω σε ίδιο στοιχείο, δεν το εισάγω. Επίσης θα γινόταν με ταξινομημένο πίνακα, τα ίδια στοιχεία θα ερχόνταν δίπλα το ένα με το άλλο οπότε με ένα πέρασμα θα μπορούσα να τα διαγράψω.

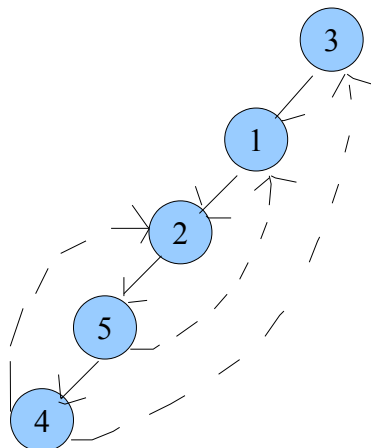
**Θέμα 3**

A)

Ο γράφος:



Το DFS: έστω ότι ξεκινάμε από την 3.



B) Βάλτε το 5 στο σύνολο S. Για κάθε γείτονα των κορυφών που ανήκουν στο σύνολο S πάλι σε αυτόν με το μικρότερο άθροισμα κοστών από τον κόμβο εκίνησης. Βάλτε τον γείτονα στο S και επανέλαβε.

Previous:	5/ 2/ 3	5	4	5	
S	1	2	3	4	5
{0}	-	-	-	-	0
{5}	20	2	-	2	
{5,2}	12		-	2	
{5,2,4}	12		4		
{5,2,4,3}	9				
<b>{5,2,4,3,1}</b>	<b>9</b>	<b>2</b>	<b>4</b>	<b>2</b>	<b>0</b>

Για να βρούμε τις διαδρομές, ελέγχουμε τα previous:

- 1: prev(1) = 3, prev(3) = 4, prev(4) = 5      5->4->3->1  
 2: prev(2) = 5      5->2  
 3: prev(3) = 4, prev(4) = 5      5->4->3  
 4: prev(4) = 5      5->4

#### Θέμα 4

A) όπως [θέμα 7A, Ιανουαρίου 2011](#)

B) HeapSort:

- 1) Αντιμετάθεση της ρίζας με το τελευταίο στοιχείο (στον πίνακα σημαίνει ότι ανταλλάσω το στοιχείο της πρώτης θέσης με το τελευταίο). Το μέγεθος του σωρού (πίνακα) μειώνεται κατά 1 γιατί το τελευταίο στοιχείο δεν θα το ξαναελέγξω, είναι στην σωστή του θέση.
- 2) Κάνω τις απαραίτητες αντιμεταθέσεις.
- 3) Επαναλαμβάνω το από το πρώτο βήμα, μέχρι ο σωρός να έχει 1 στοιχείο.

Το κόστος είναι:  $O(n)$  για την κατασκευή του σωρού +  $O(n \log n)$  για τις αντιμεταθέσεις =  $O(n \log n)$  (ο όρος που αυξάνει πιο γρήγορα).

Worst cases:

- BubbleSort:  $n^2$   
 InsertionSort:  $n^2$   
 SelectionSort:  $n^2$   
 QuickSort:  $n^2$   
 HeapSort:  $n \log n$   
 MergeSort:  $n \log n$

#### Θέμα 5

A) όπως [θέμα 5A, Ιανουαρίου 2011](#)

B) Εάν μας έδιναν δέντρο έκφρασης, θα τρέχαμε postorder. I guess.